Algorithmen für schwierige Probleme

Britta Dorn

Wintersemester 2011/12

23. November 2011

Dynamisches Programmieren

Grundidee

Speichere Lösung von kleinen Teilproblemen in Tabelle und benutze sie immer wieder, anstatt sie bei jedem rekursiven Aufruf neu zu berechnen.

Dynamisches Programmieren

Wann geht's?

Eigenschaften des Problems

1. **Optimale Teilstruktur** (optimal substructure, Bellman'sches Optimalitätsprinzip):
   Eine optimale Lösung setzt sich aus optimalen Lösungen der Teilprobleme zusammen.
2. **Sich überlappende Teilprobleme**:
   Dasselbe Problem taucht als Teilproblem von verschiedenen Problemen wieder auf. (Wird also bei Rekursion dann mehr als einmal berechnet.)
3. **Unabhängigkeit**:
   Die Lösung eines Teilproblems beeinflusst nicht die Lösung eines anderen Teilproblems desselben Problems.

Dynamisches Programmieren

4 Schritte

1. Charakterisiere die Struktur einer optimalen Lösung.
2. Gebe eine Rekursion zur Berechnung einer optimalen Lösung an.
3. Berechne den Wert einer optimalen Lösung "bottom-up" (mit kleinsten Teilproblemen starten, daraus Gesamtlösung zusammenbauen, im Gegensatz zu "top-down" (mit großem ganzen Problem starten, rekursive Aufrufe auf kleinere Probleme)).
4. Konstruiere optimale Lösung aus der bereits berechneten (und in Tabelle gespeicherten) Information (Lösung der Teilprobleme).

Farbkodierung

Viele Graphprobleme sind Spezialfälle des TEILGRAPH ISOMORPHIE Problems:

TEILGRAPH ISOMORPHIE

gegeben: Zwei Graphen $G = (V, E)$ und $G' = (V', E')$
Frage: Gibt es einen Teilgraphen von G, der zu G' isomorph ist?

(isomorph: Es gibt bij. Fkt. $f : V \to V'$ mit $\{u, v\} \in E \Leftrightarrow \{f(u), f(v)\} \in E'$)

Beispiele:

INDEPENDENT SET: $G' =$ kantenloser Graph mit k Knoten.

CLIQUE: $G' =$ vollständiger Graph mit k Knoten.

Farbkodierung

Methode für randomisierte FPT-Algorithmen für Teilgraph Isomorphie Probleme.

Farbkodierung — Beispiel Longest Path

Longest Path

gegeben: $G = (V, E)$ und $k \in \mathbb{N}$.
Frage: Gibt es einen **einfachen** Pfad der Länge k in G, d.h., Pfad aus k Knoten, so dass kein Knoten mehrfach im Pfad vorkommt?

Bemerkung: kann $k$te Potenz der Adjazenzmatrix bilden:
Eintrag $\neq 0$: Knoten sind durch Pfad der Länge k verbunden.
Problem: Pfad ist nicht unbedingt einfach.

Deshalb jetzt mit Farbkodierung!

Farbkodierung — Beispiel LONGEST PATH

Grundidee (randomisierter Ansatz)

Färbe Knoten von G per Zufall mit k verschiedenen Farben. Hoffe, dass alle Knoten eines Pfades unterschiedliche Farben bekommen.

Wir nennen einen Pfad *vollbunt*, wenn jeder seiner Knoten mit anderen Farbe gefärbt wurde.

Klar: Pfad voll bunt $\Rightarrow$ Pfad ist einfach.

Was lässt sich für einfache Pfade sagen? Vollbunt??